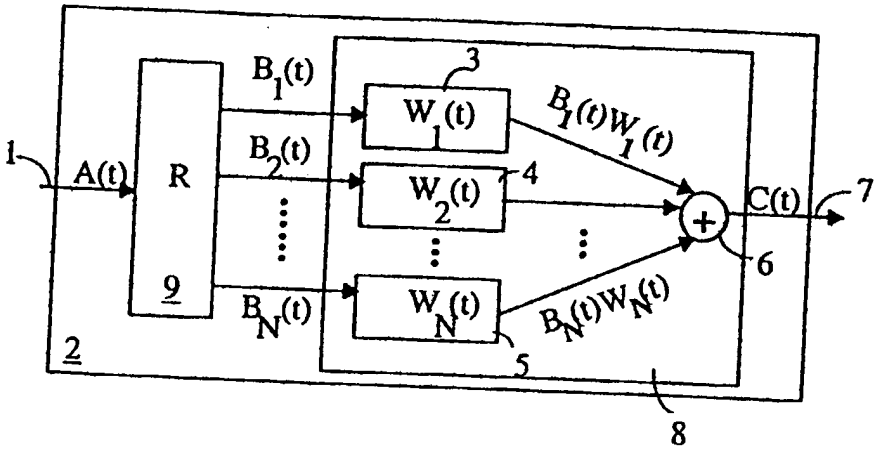

FIG. 1A

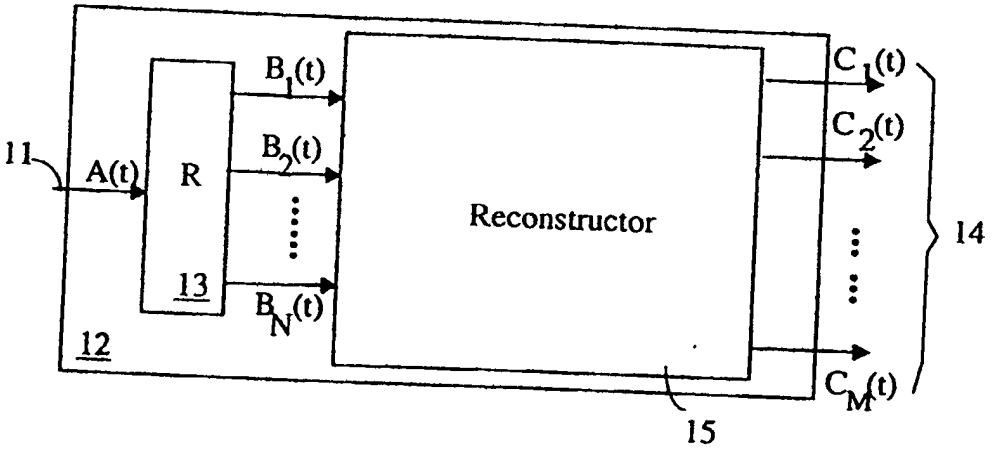

FIG. 1B

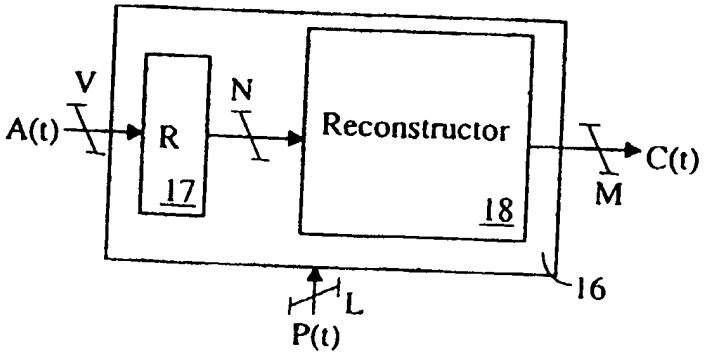

FIG. 1C

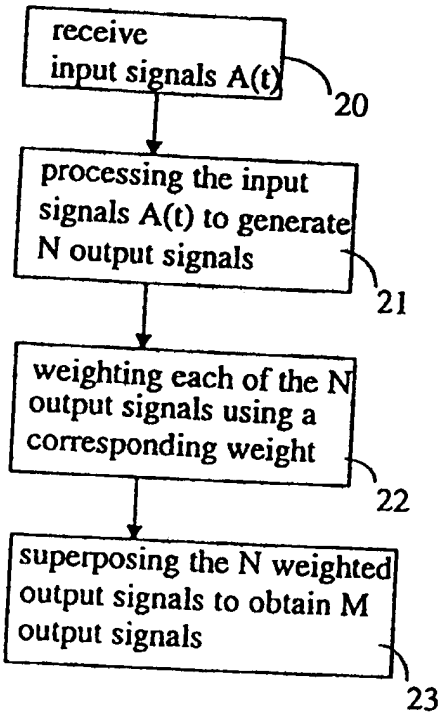

FIG. 2

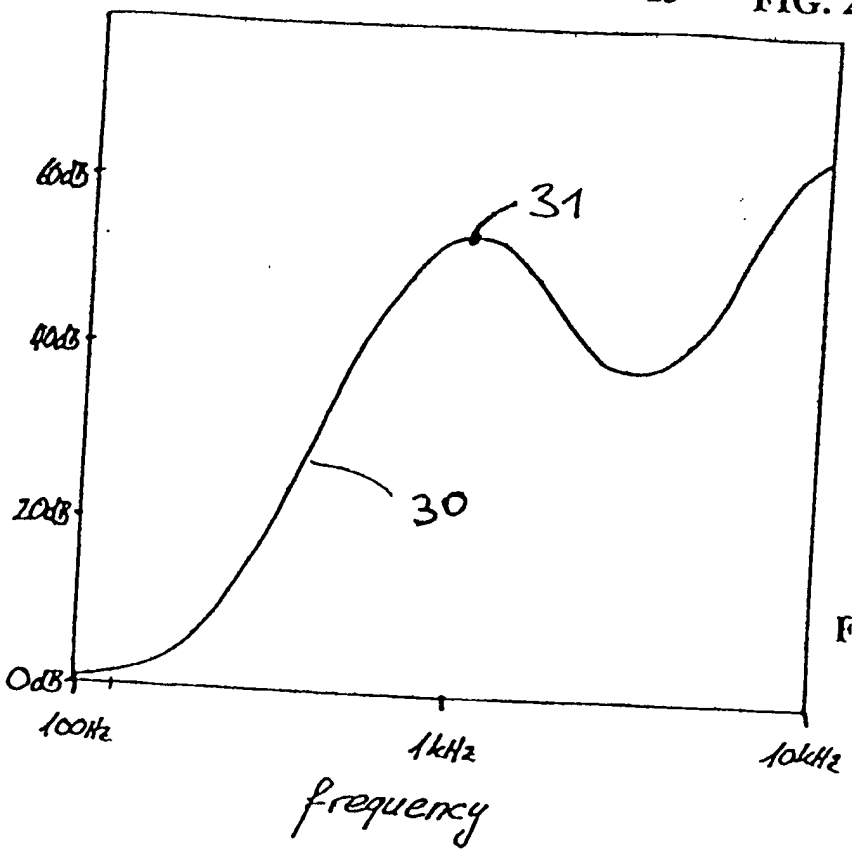

FIG. 3

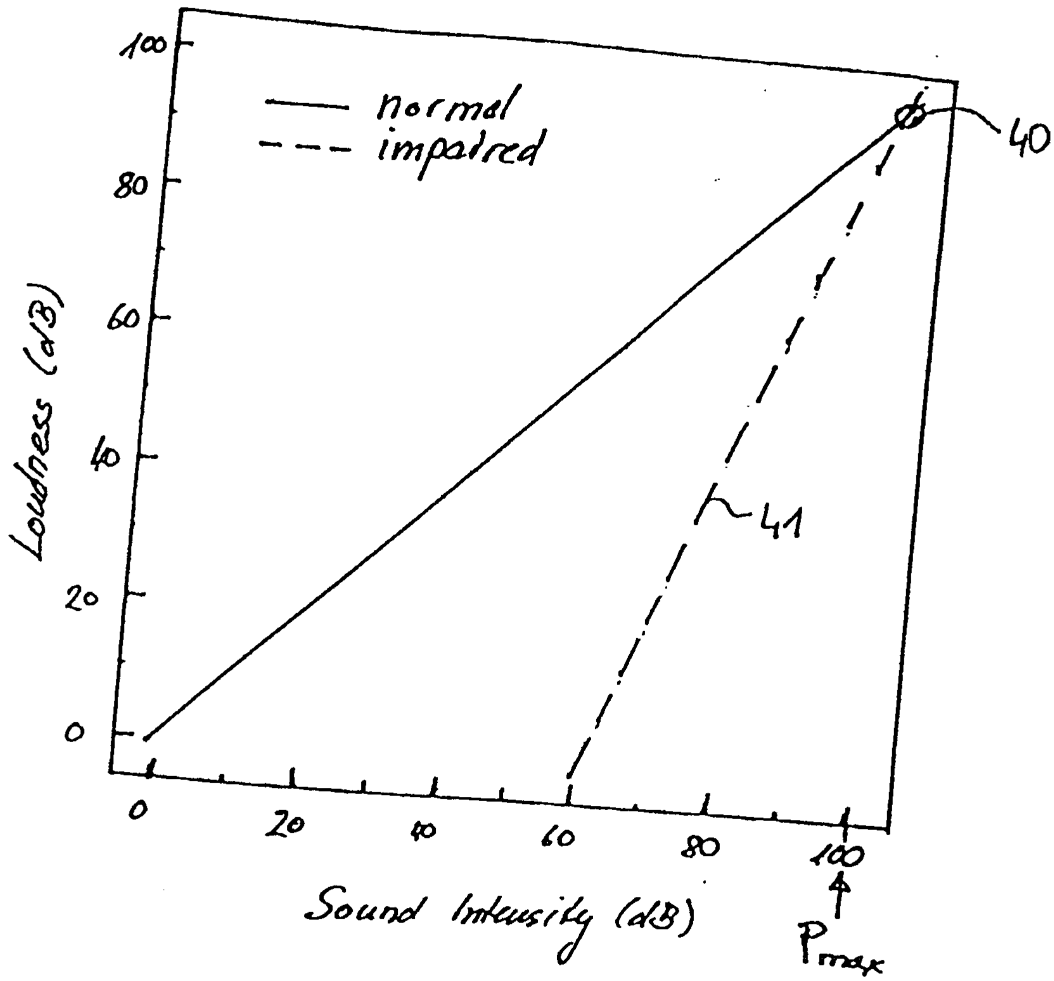

FIG. 4

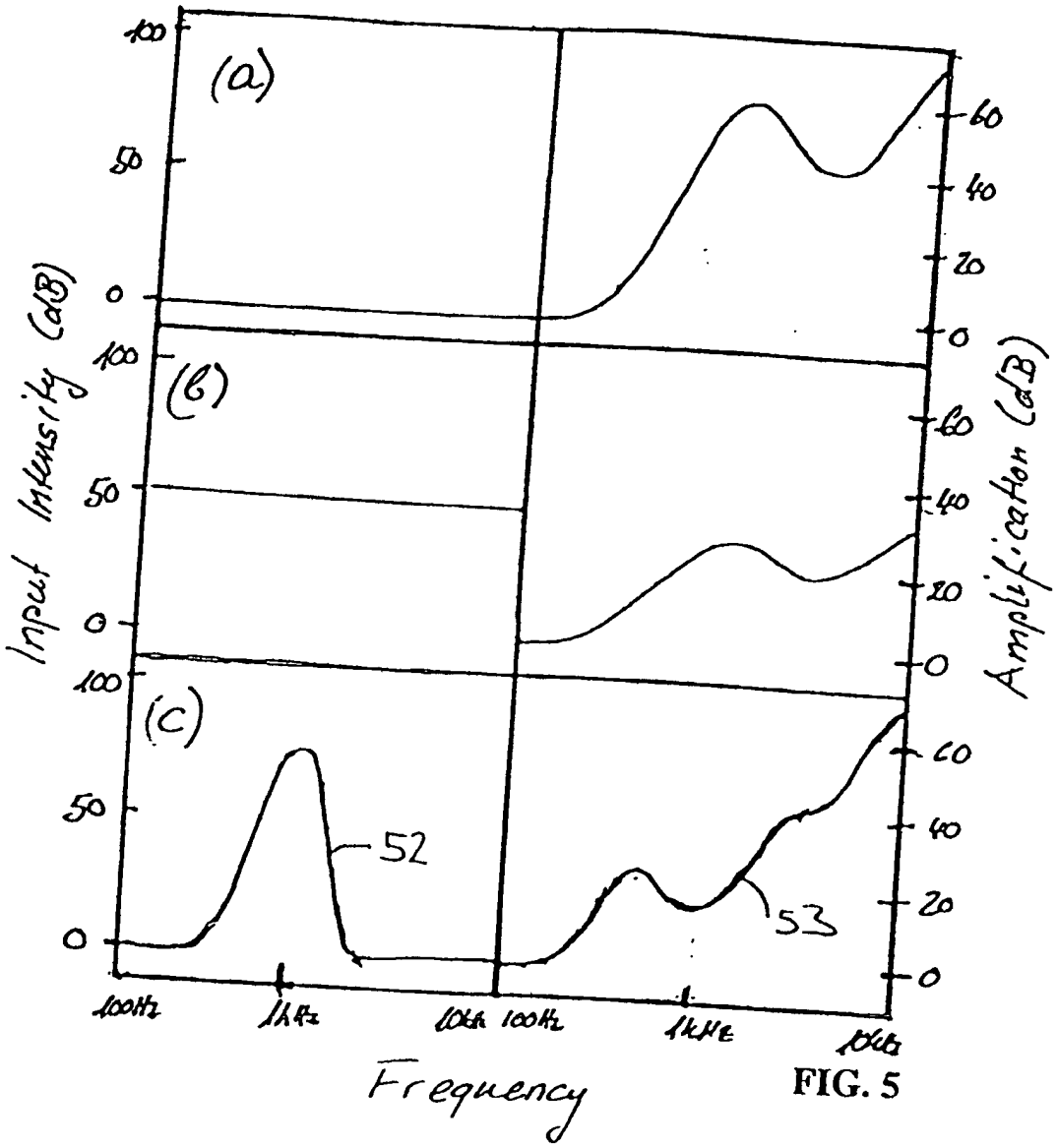
(a)
(b)
(c)
100
50
0
60
40
20
0
Input Intensity (dB)
Amplification (dB)
52
53
100Hz
1kHz
10kHz
100Hz
1kHz
10kHz
Frequency

FIG. 5

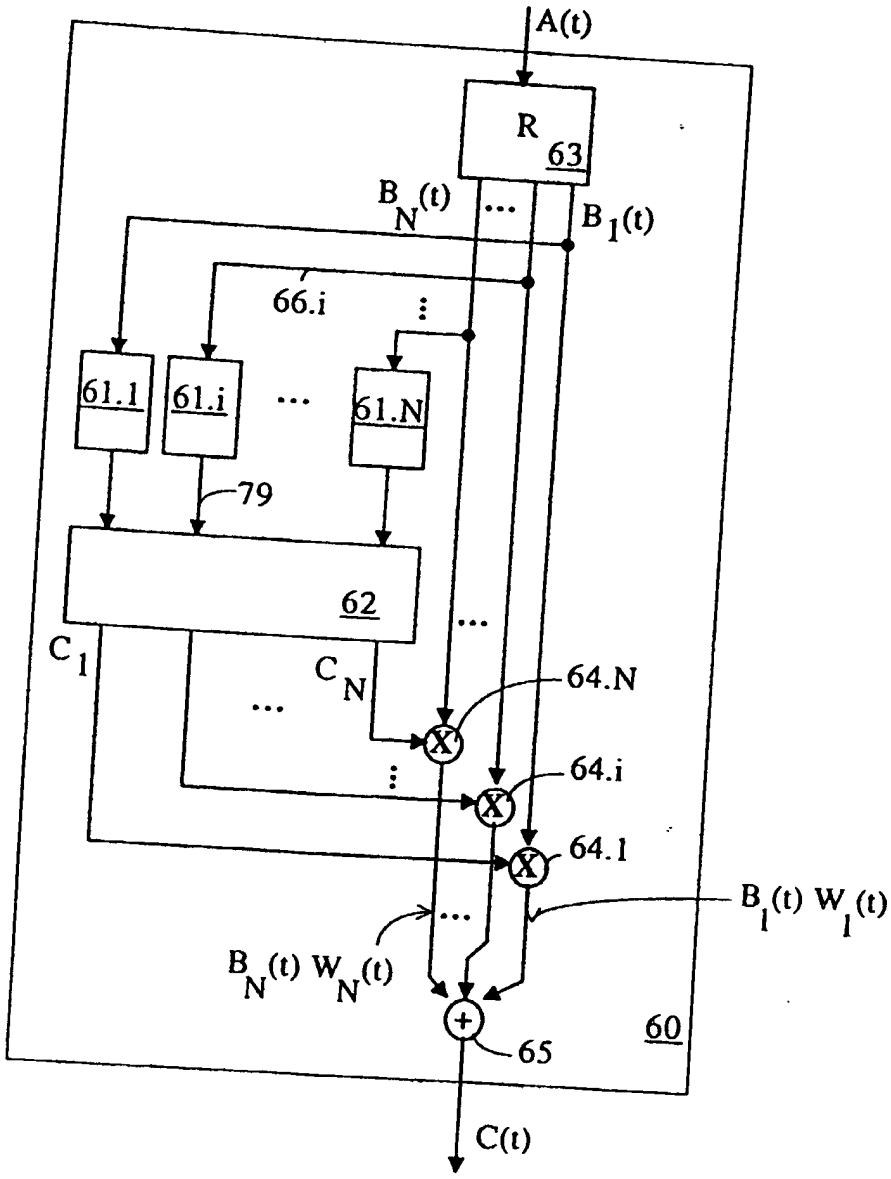
A(t)
R
63
$B_N(t)$
...
$B_1(t)$
66.i
61.1
61.i
61.N
79
62
C_1
C_N
64.N
64.i
64.1
$B_l(t)\ W_l(t)$
$B_N(t)\ W_N(t)$
65
60
C(t)

FIG. 6

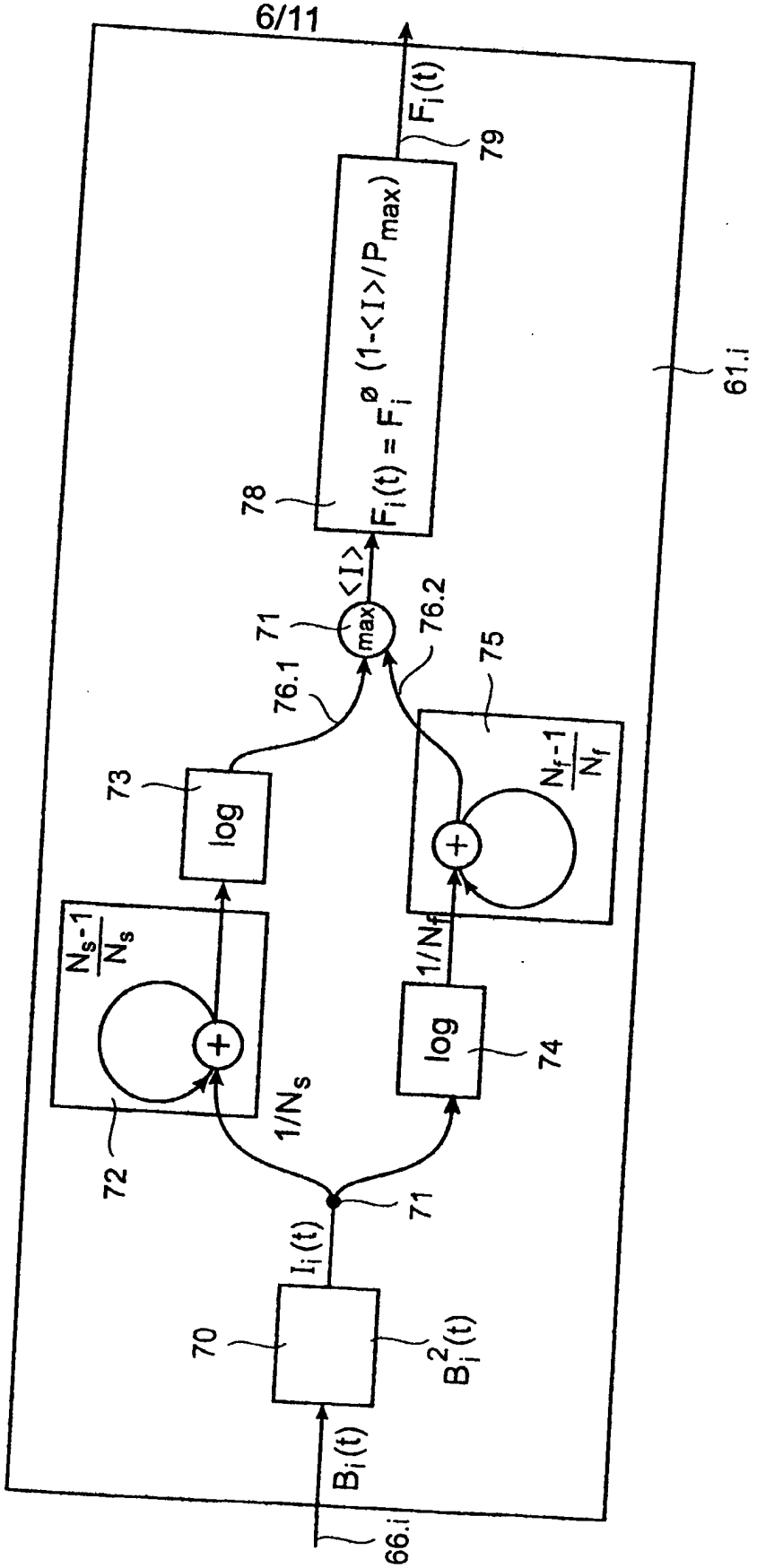

Fig. 7

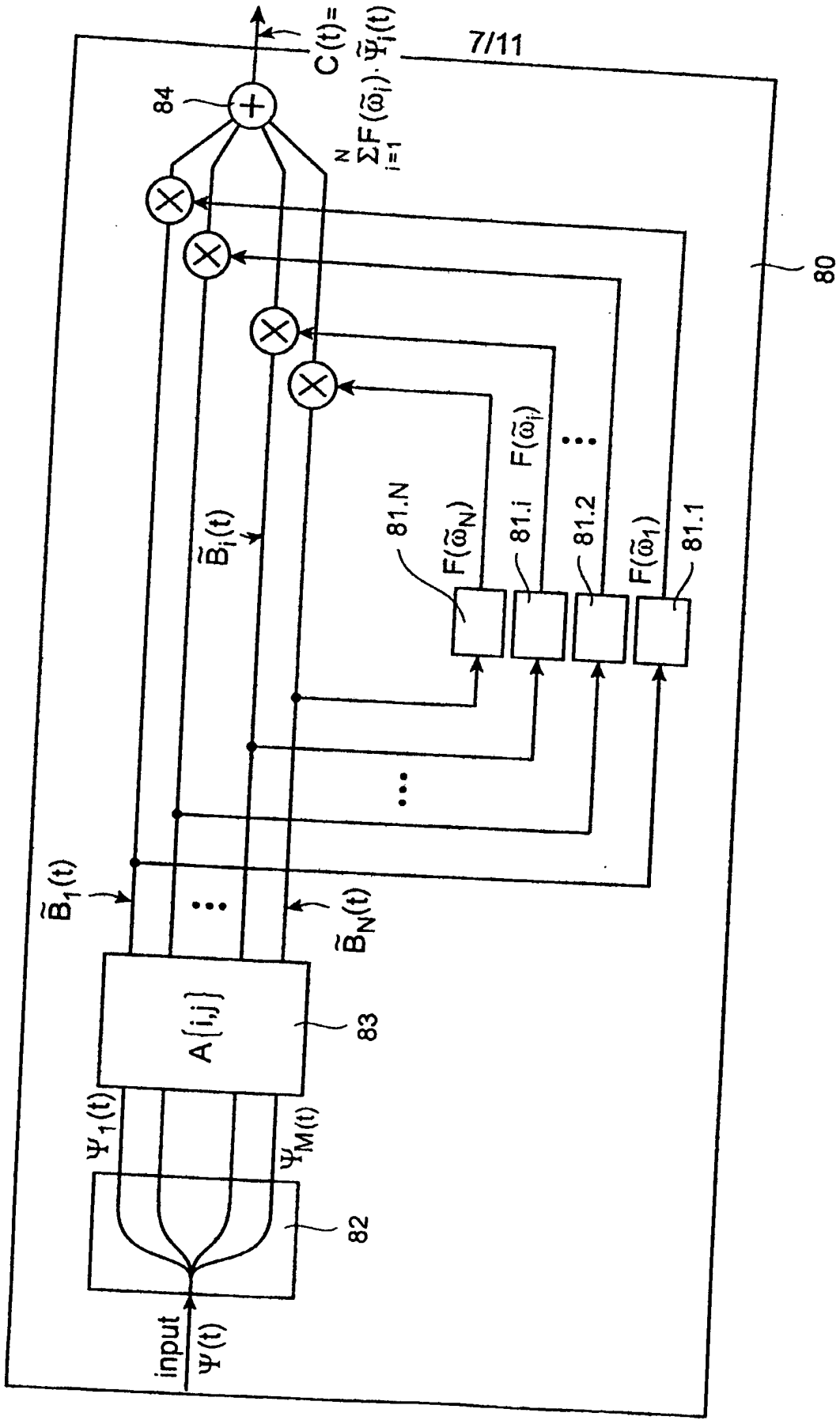

Fig. 8

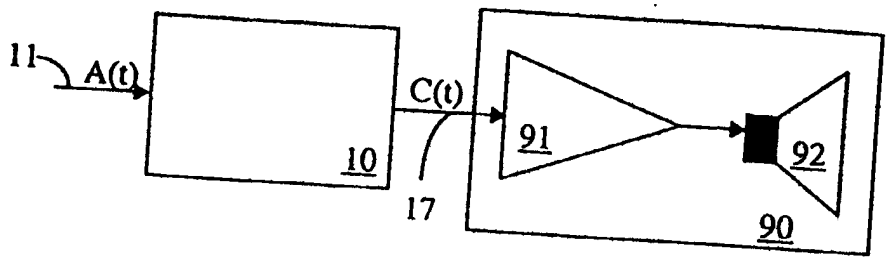

FIG. 9

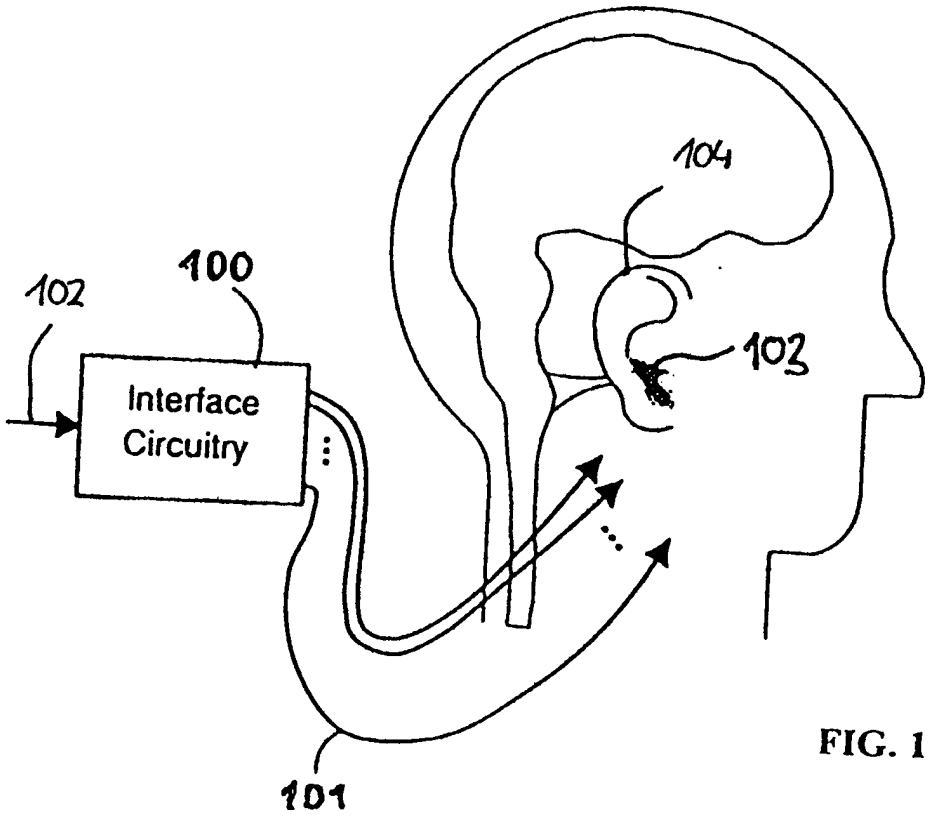

FIG. 10

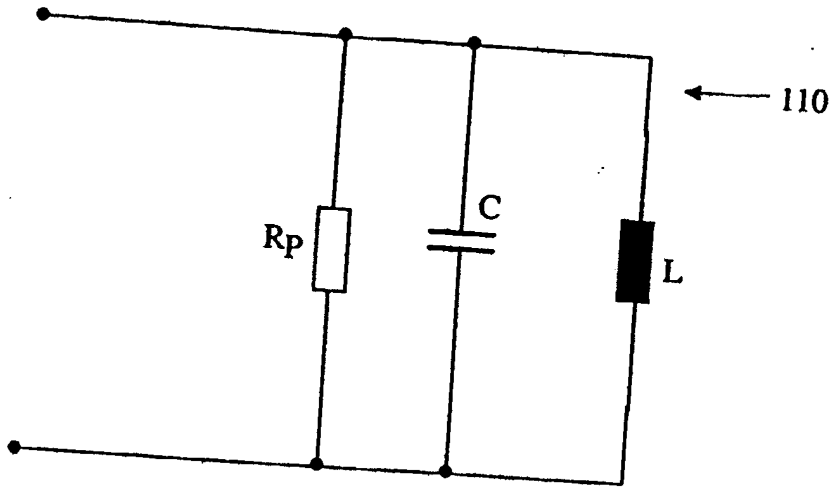

FIG. 11

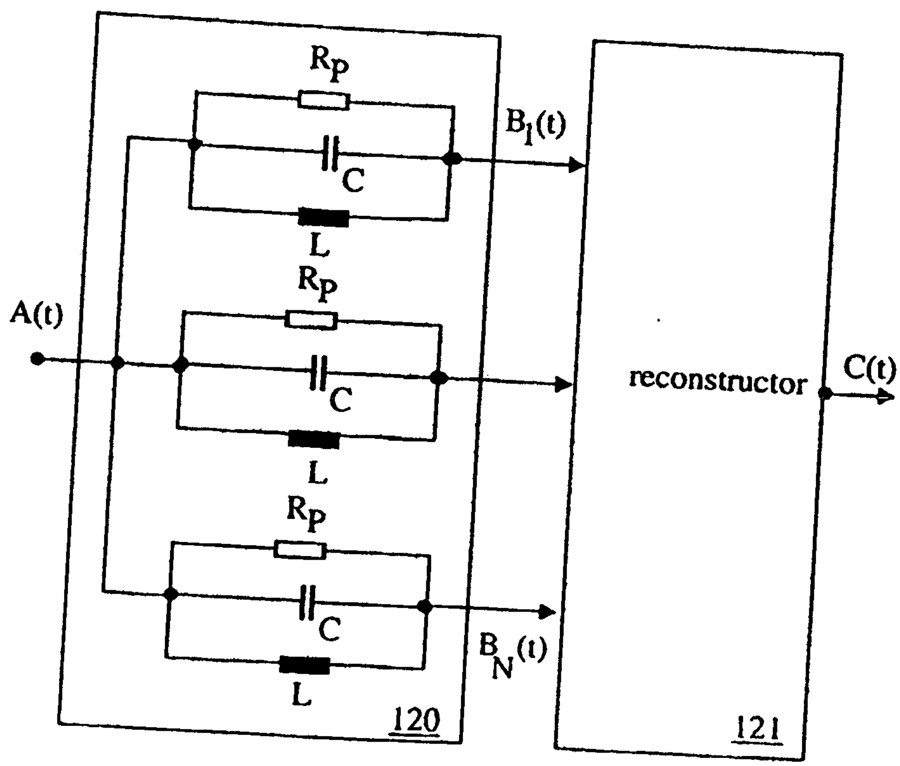

FIG. 12

FIG. 13A

```
Implementation example:
=======================================

[read discretization time step]

get dt

[reset oscillators]

130  for each N                          [loop over resonators]

       get omega(N)    [read frequency *2*pi]
       get tau(N)      [read time decay constant]
       get weight(N)   [read weight]

       u0(N)= (2-(omega(N)dt)²) / (1+dt/tau(N))
                                  [one example of equations describing
                                  [a damped harmonic resonator]
       um(N)= -(1 - dt/tau(N)) / (1+dt/tau(N))
       ui(N)= dt / (1+dt/tau(N)) / tau(N)
                                  [u are constants]

       bm(N)=0                    [b are amplitudes of resonators]
       b0(N)=0                   [bm is amplitude of previous time step]
131                               [bo is amplitude of actual time step]
     done with loop over N

     [loop over time steps]
132
     for each t
       input ap                   [read input signal a of the]
                                  [next time step]

       if(first t) then           [the first time, reset stack]
                                  [for input; otherwise a would]
135      a0=ap                     [not be defined]
         am=ap
       end if

     [calculate difference aprime of input signal a]

       aprime = (ap-am)

     continue with FIG. 13B
```

FIG. 13B

```
continuation of FIG. 13A

[propagate resonators; loop over all resonators]

  for each N
134   b_p = u_0(N)*b_0(N) + u_m(N)*b_m(N) + u_i(N)*aprime
                                  [resonator for following time]
    b_m(N) = b_0(N)               [step]
    b_0(N) = b_p                  [stack is updated/shifted]
  done with loop over N


[at this point the weights weight(N) can be recalculated]
[as a function of the b_0(N)]
[In this implementation example the weights are constant]

[weight and combine resonator output signals to generate]
[output signal c]

    c=0
    for each N
      c = c + weight(N)*b_0(N)
    done with loop over N

[reset stack for input]
    a_m = a_0
    a_0 = a_p

[output output signal c]

    output c

133 done with loop over t

end.
=================================
```